NUMERICAL ASPECTS OF $\kappa - \epsilon$ TURBULENCE MODELING USING A FINITE ELEMENT INCOMPRESSIBLE NAVIER-STOKES FORMULATION

Norberto Nigro, Mario Storti and Angel Zanotti

International Center for Computational Methods in Engineering (CIMEC)
INTEC-Universidad Nacional del Litoral-CONICET, Güemes 3450
3000 Santa Fe, Argentina
e-mail: mstorti, Web page: http://venus.arcride.edu.ar/CIMEC)

Key Words: incompressible Navier-Stokes equations, $\kappa - \epsilon$ turbulence model

Abstract. *In this work we present a stabilized equal order finite element formulation of incompressible Navier-Stokes equations augmented by a $\kappa - \epsilon$ turbulence model. The aim of this paper is to evaluate the main numerical difficulties associated with the solution of this kind of problems, mainly the possitiveness of the mathematical operators involved and the rate of convergence of the whole system. We propose a particular way to circumvent these drawbacks using*

- *.- an extra stabilization term in the transport equations of the turbulence quantities to avoid undershoots in the $\kappa - \epsilon$ fields,*
- *.- a smooth enough cutoff function to avoid negative values of $\kappa - \epsilon$ fields, and*
- *.- an almost fully implicit monolithic solution strategy in order to reach good convergence rates of the whole system.*

The incompressible Navier-Stokes equations are spatially discretized by a SUPG-PSPG technique and temporally solved by a backward Euler scheme.

This work was done as part of our project regarding to the implementation of PETSc-FEM code (http://minerva.arcride.edu.ar/petscfem/petscfem), a general purpose, multiphysics library running on Beowulf (Intel processors+Unix/Linux OS) cluster[2] *and based on the MPI message passing library*[1] *and the Parallel Extensible Toolkit for Scientific Computations (PETSc),*[3] *written in object oriented programming using C++.*

1 INTRODUCTION

In order to fix ideas, we focus on the solution of the incompressible Navier-Stokes equations with the SUPG+PSPG method proposed by Tezduyar et.al.[4] The Navier-Stokes equations present two important difficulties for the solution with finite elements. First, the character of the equation becomes more and more advective dominant when the Reynolds number increases. In addition the incompressibility condition represents not an evolution equation but, rather, a constraint on the equations. This has the drawback that only some combination of interpolation spaces for velocity and pressure can be used, namely those ones that satisfy the so called Brezzi-Babŭska condition. In the formulation of Tezduyar et.al.[4] advection is stabilized with the well known SUPG stabilization term, and a similar stabilization term called PSPG is included in order to circumvent checkerboard modes. Once these equations are discretized, the resulting system of ODE's are discretized in time with the standard backward Euler scheme. At every time step, the resulting non-linear system of equations is solved iteratively with the GMRES method with Jacobi right preconditioning.

2 MATHEMATICAL MODELING OF TURBULENT INCOMPRESSIBLE FLUID FLOW

This section deals with the mathematical modeling of the turbulent incompressible fluid flow that is organized in the following form: it starts with the mathematical description of the continuity and momentum equations, follows with the description of the transport equations for the $\kappa - \epsilon$ turbulence model and finishs with the wall law function used to represent the flow field in the vecinity of rigid walls.

2.1 Mass and momentum equations

The continuity and momentum balance equations for laminar flows can be written in the following form:

$$\begin{aligned} \nabla \cdot \boldsymbol{u} = \mathbf{o} \ \text{ in } \ \mathit{\Omega} \times (0,T) \\ \rho(\frac{\partial \boldsymbol{u}}{\partial t} + \boldsymbol{u} \cdot \nabla \boldsymbol{u}) - \nabla \cdot \boldsymbol{\sigma} = \mathbf{o} \ \text{ in } \ \mathit{\Omega} \times (0,T), \end{aligned} \tag{1}$$

with ρ and $\boldsymbol{u}$ the density and velocity of the fluid and $\boldsymbol{\sigma}$ the stress tensor, given by

$$\begin{aligned} \boldsymbol{\sigma} &= -p\boldsymbol{I} + 2\mu_{eff}\boldsymbol{\epsilon}(\boldsymbol{u}) \\ \boldsymbol{\epsilon}(\boldsymbol{u}) &= 1/2(\nabla \boldsymbol{u} + (\nabla \boldsymbol{u})^t) \end{aligned} \tag{2}$$

where p and $\mu_{eff} = \rho\nu_{eff}$ are pressure and the effective dynamic viscosity proportional to the effective kinematic viscosity defined below, $\boldsymbol{I}$ represents the identity matrix and $\boldsymbol{\epsilon}$ the deformation tensor.

2.2 Two equation turbulence modeling

In this paper the turbulent effects are included via a $\kappa-\epsilon$ model, which is based on the transport of two additional turbulent quantities, the turbulent kinetic energy κ and its dissipation rate ϵ. These two additional balance equations may be written in the following form:

$$\begin{aligned}
(\frac{\partial\kappa}{\partial t}+\boldsymbol{u}\cdot\nabla\kappa) &= \nabla\cdot(\frac{\nu_{eff}}{C_\kappa}\nabla\kappa)+P_\kappa-\epsilon \\
(\frac{\partial\epsilon}{\partial t}+\boldsymbol{u}\cdot\nabla\epsilon) &= \nabla\cdot(\frac{\nu_{eff}}{C_\epsilon}\nabla\epsilon)+\frac{\epsilon}{\kappa}(C_1P_\kappa-C_2\epsilon)
\end{aligned} \tag{3}$$

where $\nu_{eff}=\nu+\nu_t$ represents the effective kinematic viscosity, sum of the molecular and the turbulent effects, being the latter defined in terms of the two added turbulent fields as:

$$\nu_t=C_\mu\frac{\kappa^2}{\epsilon} \tag{4}$$

In 3 each transport equation has a standard mathematical pattern, a temporal term represented by a $\frac{\partial}{\partial t}$ symbol, a convective term defined by a first order spatial operator ∇, a diffusive term proportional to a second order spatial operator $\nabla\nabla$ and source terms. The incompressible Navier-Stokes equations and the $\kappa-\epsilon$ model just described are nonlinear and coupled.

The source terms in 3 are written in terms of P_κ, defined as :

$$P_\kappa=2\nu_t\boldsymbol{\epsilon}(\boldsymbol{u}):\boldsymbol{\epsilon}(\boldsymbol{u}) \tag{5}$$

The $\kappa-\epsilon$ model has five constants,

$$\begin{aligned}
C_\mu &= 0.09 \\
C_1 &= 1.44 \\
C_2 &= 1.92 \\
C_\kappa &= 1.0 \\
C_\epsilon &= 1.3
\end{aligned} \tag{6}$$

The continuum formulation is completed by the initial and boundary conditions. Relative to the boundary conditions, unless for the pressure, we may split the whole boundary in parts, imposing in each part a Dirichlet condition, a Neumann condition or a near wall boundary condition. For the velocity field we have

$$\begin{aligned} \mathit{\Gamma} &= \mathit{\Gamma}_g \cup \mathit{\Gamma}_h \cup \mathit{\Gamma}_{wall} \\ \mathit{\Gamma}_g \cap \mathit{\Gamma}_h \cap \mathit{\Gamma}_{wall} &= \emptyset \\ \boldsymbol{u} &= \boldsymbol{g} \qquad \text{on } \mathit{\Gamma}_g \\ \boldsymbol{n} \cdot \boldsymbol{\sigma} &= \boldsymbol{h} \qquad \text{on } \mathit{\Gamma}_h \\ \boldsymbol{n} \cdot \boldsymbol{\sigma} &= \boldsymbol{h}_{wall}(u_*(\boldsymbol{u})) \qquad \text{on } \mathit{\Gamma}_{wall} \end{aligned} \tag{7}$$

For the turbulent fields the boundary conditions may be written as:

$$\begin{aligned} \mathit{\Gamma} &= \mathit{\Gamma}_g \cup \mathit{\Gamma}_h \cup \mathit{\Gamma}_{wall} \\ \mathit{\Gamma}_g \cap \mathit{\Gamma}_h \cap \mathit{\Gamma}_{wall} &= \emptyset \\ \phi &= g \qquad \text{on } \mathit{\Gamma}_g \\ \boldsymbol{n} \cdot \nabla\phi &= h \qquad \text{on } \mathit{\Gamma}_H \\ \phi &= \phi_{wall}(u_*(\boldsymbol{u})) \qquad \text{on } \mathit{\Gamma}_{wall} \end{aligned} \tag{8}$$

with $\phi = \kappa$ or $\phi = \epsilon$.

In 7 and 8 u_* is the wall friction velocity that is computed using the wall law presented below.

The pressure should be fixed at a reference value in at least one node in order to remove the corresponding rigid mode in the numerical computation.

For the initial conditions we initialize the computation with a particular field for each unknown that depends on the simulation.

2.3 Wall law function

In this section we describe how to compute the wall friction velocity needed to fully define the boundary conditions of the momentum and turbulent transport equations.

In 7 and 8 $\mathit{\Gamma}_{wall}$ represents that part of the boundary where wall law function is imposed. Through this boundary condition the wall traction for the momentum equations and the values of κ and ϵ close to the wall are computed. This procedure involves the computation of the wall friction velocity u_* whose profile is assumed to be represented by the following expressions:

$$\begin{aligned} \frac{|\boldsymbol{u} - \boldsymbol{u}_{wall}|}{u_*} &= f(y^+) = \begin{cases} y^+ & \text{for } \; y^+ < 5 \text{ laminar region} \\ 5\log(y^+) + C_{wall,1} & \text{for } \; 5 < y^+ < 30 \text{ buffer region} \\ 2.5\log(y^+) + C_{wall,2} & \text{for } \; y^+ > 30 \text{ full logarithmic region} \end{cases} \\ y^+ &= \frac{y_{wall}\, u_*}{\nu} \end{aligned} \tag{9}$$

where y_{wall} is the distance between the wall and the first node close to the wall lying over the computational boundary. For solving the nonlinear equation associated with the computation of u_* we have used a secant method. Once u_* is computed the traction for the momentum equation is determined by:

$$\begin{aligned} \boldsymbol{h}_{wall} &= g(\boldsymbol{u} - \boldsymbol{u}_{wall}) \\ g &= \frac{|\tau_{wall}|}{|\boldsymbol{u} - \boldsymbol{u}_{wall}|} \\ |\tau_{wall}| &= \rho u_*^2 \end{aligned} \tag{10}$$

Finally the corresponding κ_{wall} and ϵ_{wall} values are obtained from:

$$\begin{aligned} \kappa_{wall} &= \frac{u_*^2}{\sqrt{C_\mu}} \\ \epsilon_{wall} &= \frac{u_*^4}{\chi y^+ \nu} \end{aligned} \tag{11}$$

with χ the von-Karman constant.

3 NUMERICAL ANALYSIS OF THE PROBLEM

This section is devoted to analyze numerically how to improve the computation of a turbulent flow field using a finite element formulation. In the next section we describe the stabilized equal order finite element method applied to incompressible Navier-Stokes equations using the standard SUPG-PSPG method. Next section deals with the numerical formulation for the two turbulent transport equations with emphasis on the numerical stabilization of the additional numerical instabilities produced when high source terms appears in some localized regions of the domain. In the following section we present some topics about improving the convergence rate of the numerical method. A coupled fully implicit method is described putting special emphasis on the treatment of the wall law function. Next section is devoted to the presentation of a smooth enough cutoff function introduced to limit the appearance of negative values in the turbulent fields. Both $\kappa < 0$ and $\epsilon < 0$ are physically meaningless, the former by definition and the latter because this situation produces negative values in the turbulent viscosity according to 4. The regularity of the cutoff function is crucial to guarantee the convergence of the algebraic system. In the last section we present the solver adopted to get the solution of the algebraic system of equations obtained from the numerical discretization of the incompressible Navier-Stokes equations .

3.1 Finite element formulation for incompressible Navier-Stokes equations

The functional spaces for the weight and interpolation functions are defined as follow:

$$
\begin{aligned}
S_{\boldsymbol{u}}^h &= \{\boldsymbol{u}^h | \boldsymbol{u}^h \in (H^{1h})^{nsd}, \boldsymbol{u}^h \doteq \boldsymbol{g}^h \ \text{ in} \Gamma_g\} \\
V_{\boldsymbol{u}}^h &= \{\boldsymbol{N}^h | \boldsymbol{N}^h \in (H^{1h})^{nsd}, \boldsymbol{N}^h \doteq \mathbf{o} \ \text{ in} \Gamma_g\} \\
S_p^h &= \{q^h | q^h \in H^{1h}\} \\
V_p^h &= \{p^h | p^h \in H^{1h}\}
\end{aligned}
\tag{12}
$$

ith

$$
H^{1h} = \left\{ \phi^h | \phi^h \in C^0(\overline{\Omega}), \phi^h|_{\Omega^e} \in P^1, \forall \Omega^e \in \mathcal{E} \right\}
\tag{13}
$$

the Sobolev space, P^1 is a first order polynomial set, nsd is the dimension of the physical domain $\Omega = \cup \Omega^e$, $\mathcal{E}$ means the discrete partition of the physical domain being Ω^e the part of this partition corresponding to the element e.

The *SUPG-PSPG* formulation of (1) is written as:

Find $\boldsymbol{u}^h \in S_u^h$ and $p^h \in S_p^h$ satisfying

$$
\begin{aligned}
&\int_\Omega \boldsymbol{N}^h \cdot \rho \Big(\frac{\partial \boldsymbol{u}^h}{\partial t} + \boldsymbol{u}^h \cdot \nabla \boldsymbol{u}^h \Big) + \int_\Omega \epsilon(\boldsymbol{N}^h) : \boldsymbol{\sigma}^h d\Omega + \\
&+ \underbrace{\sum_{e=1}^{nel} \int_\Omega \boldsymbol{\delta}^h \cdot \Big[\rho (\frac{\partial \boldsymbol{u}^h}{\partial t} + \boldsymbol{u}^h \cdot \nabla \boldsymbol{u}^h) - \nabla \cdot \boldsymbol{\sigma}^h \Big]}_{(SUPG)} d\Omega + \\
&+ \underbrace{\sum_{e=1}^{nel} \int_\Omega \boldsymbol{\epsilon}^h \cdot \Big[\rho (\frac{\partial \boldsymbol{u}^h}{\partial t} + \boldsymbol{u}^h \cdot \nabla \boldsymbol{u}^h) - \nabla \cdot \boldsymbol{\sigma}^h \Big]}_{(PSPG)} d\Omega + \\
&+ \sum_{e=1}^{nel} \int_\Omega \tau_{(CONT)} \nabla \cdot \boldsymbol{N}^h \rho \nabla \cdot \boldsymbol{u}^h d\Omega + \int_\Omega q^h \nabla \cdot \boldsymbol{u}^h d\Omega = \int_{\Gamma_h} \boldsymbol{N}^h \cdot \boldsymbol{h}^h d\Gamma \qquad \forall \boldsymbol{N}^h \in V_u^h, \forall q^h \in V_p^h
\end{aligned}
\tag{14}
$$

The stabilization parameters are defined as:

$$
\begin{aligned}
\boldsymbol{\delta}^h &= \tau_{\mathrm{s}upg}(\boldsymbol{u}^h \cdot \nabla)\boldsymbol{N}^h\,, \\
\boldsymbol{\epsilon}^h &= \tau_{\mathrm{p}spg}\frac{1}{\rho}\nabla q^h \ \ , \\
\tau_{\mathrm{S}UPG} &= \frac{h}{2||\boldsymbol{u}^h||} z(Re_u) \\
\tau_{\mathrm{P}SPG} &= \tau_{\mathrm{S}UPG} \\
\tau_{\mathrm{C}ONT} &= \frac{h}{2}||\boldsymbol{u}^h|| z(Re_u)
\end{aligned}
\tag{15}
$$

with Re_u and Re_U are the Reynolds numbers based on the element parameters, i.e.:

$$
Re_u = \frac{||\boldsymbol{u}^h||h}{2\nu} \tag{16}
$$

The element length h is computed by:

$$
h = 2\Big(\sum_{a=1}^{nen} |\boldsymbol{s} \cdot \nabla w_a|\Big)^{-1} \tag{17}
$$

with w_a the shape function associated with the a node, nen the number of nodes in the element, and $\boldsymbol{s}$ a unit normalized velocity vector. The function $z(Re)$ used in (16) is defined as:

$$
z(Re) = \begin{cases} Re/3 & 0 \leq Re < 3, \\ 1 & 3 \leq Re \end{cases} \tag{18}
$$

The spatial discretization leads to the following system of equations:

$$
\begin{aligned}
(\boldsymbol{M} + \boldsymbol{M}_\delta)\boldsymbol{a} + \boldsymbol{N}(\boldsymbol{v}) + \boldsymbol{N}_\delta(\boldsymbol{v}) + (\boldsymbol{K} + \boldsymbol{K}_\delta)\boldsymbol{v} - (\boldsymbol{G} - \boldsymbol{G}_\delta)\boldsymbol{p} &= \boldsymbol{F} + \boldsymbol{F}_\delta \\
\boldsymbol{G}^t\boldsymbol{v} + \boldsymbol{M}_\epsilon\boldsymbol{a} + \boldsymbol{N}_\epsilon(\boldsymbol{v}) + \boldsymbol{K}_\epsilon\boldsymbol{v} + \boldsymbol{G}_\epsilon\boldsymbol{p} &= \boldsymbol{E} + \boldsymbol{E}_\epsilon
\end{aligned}
\tag{19}
$$

3.2 Guaranting Possitiveness in the numerical scheme for the $\kappa - \epsilon$ fields

As we have mentioned above the $\kappa - \epsilon$ transport equations are generic transient advection-diffusion systems with source terms dependent of the state variables. In general terms we may write 3 as:

$$
(\frac{\partial \phi}{\partial t} + \boldsymbol{u} \cdot \nabla\phi) = \nabla \cdot (\frac{\nu_{eff}}{C_\phi}\nabla\phi) + \mathcal{S}_\phi \tag{20}
$$

with $\phi = \kappa$ or $\phi = \epsilon$. C_ϕ and $\mathcal{S}_\phi$ depend on ϕ selected. Even though these two equations are coupled through of ν_{eff} and $\mathcal{S}_\phi$ we write them as an uncoupled system in order to present the numerical method used to discretize them. We wish to remark that the implementation of mass, momentum and turbulent equations is fully coupled unless for the stabilization parameters in order to simplify the computation. Before enter into details about the numerical method used we wish to do a special remark about to 20 equation. As the source term $\mathcal{S}_\phi$ depends on the state variable we can redefine it as:

$$\mathcal{S}_\phi = c(\phi)\phi + \mathcal{S}'_\phi \tag{21}$$

Rewriting 20 with 21 we arrive to the following equation:

$$(\frac{\partial \phi}{\partial t} + \boldsymbol{u} \cdot \nabla \phi) - c(\phi)\phi = \nabla \cdot (\frac{\nu_{eff}}{C_\phi} \nabla \phi) + \mathcal{S}'_\phi \tag{22}$$

that is similar to a generic unsteady reaction-advection-diffusion system. Two dimensionless parameters may be built from this equation, the standard Peclet number (Pe) and other called the reaction number (Rn) defined as:

$$\begin{aligned} Pe &= \frac{\boldsymbol{u} h_{\text{grid}}}{2\frac{\nu_{eff}}{C_\phi}} \\ Rn &= \frac{c h_{\text{grid}}^2}{\frac{\nu_{eff}}{C_\phi}} \end{aligned} \tag{23}$$

It is well known that numerical oscillations are found when both Pe and Re number grows above their critical values.

In order to get a numerical solution of 20 or 22 we have adopted an SUPG strategy enhanced by the possibility of avoiding the undershoots and overshoots associated with high reaction number values. In the past we have designed a special perturbation function to be added to the standard perturbation function of the SUPG method with very good results but in that opportunity we focussed only on the scalar case.[7] This kind of strategy may be applied to each equation separately or may be extended to the case of vectorial state variables but we leave this kind of analysis for a future work. In this work we have chosen a very simple solution generally used to overcome some localized undershoot and/or overshoots frequently found close to discontinuities. This strategy consists of adding an extra isotropic numerical diffusion localized only where the reaction number dominates over Peclet number.

$$
\begin{aligned}
\int_{\Omega} \tilde{w}^h \Big(\frac{\partial \phi^h}{\partial t} + \boldsymbol{u}^h \cdot \nabla \phi^h \Big) + \int_{\Omega} \nabla w^h \frac{\nu_{eff}}{C_\phi} \nabla \phi^h d\Omega - \int_{\Omega} \tilde{w}^h \mathcal{S}_\phi + \\
\underbrace{\int_{\Omega} \nabla w^h \delta_\phi \nabla \phi^h d\Omega}_{\text{extra isotropic numerical diffusion}} \quad - \int_{\Gamma_h} w^h h^h d\Gamma = 0
\end{aligned}
\tag{24}
$$

The weight function in SUPG and the main stabilitization parameter are generally defined as:

$$
\begin{aligned}
\tilde{w}^h &= w^h + P^h \\
P^h &= \tau \boldsymbol{u}^h \nabla \cdot w^h \\
\tau &= \frac{h_{\mathrm{g}rid}}{2|\boldsymbol{u}^h|} \psi(Pe) \\
\psi(Pe) &= \coth(Pe) - \frac{1}{Pe}
\end{aligned}
\tag{25}
$$

In general the second stabilization parameter in the extra isotropic numerical diffusion is defined as:

$$
\delta_\phi = Rn \, \frac{\nu_{eff}}{C_\phi} \, \psi(Rn) \tag{26}
$$

- Reaction numbers definition for κ and ϵ

Specifically for the $\kappa - \epsilon$ model the second stabilization parameter for each one of the turbulent transport equations is defined as:

$$
\begin{aligned}
\delta_\kappa &= \lambda_{max} \psi(\lambda_{max}) \frac{\nu_{eff}}{C_\kappa} \\
\delta_\epsilon &= \lambda_{max} \psi(\lambda_{max}) \frac{\nu_{eff}}{C_\epsilon} \\
\lambda_{max} &= h_{\mathrm{g}rid}^2 \max\{\max\{\lambda_1, \lambda_2\}, \varepsilon\}
\end{aligned}
\tag{27}
$$

where $\lambda_{1,2}$ are computed by:

$$
\begin{aligned}
C_{\kappa,\epsilon} &= \begin{pmatrix} \frac{\partial \mathcal{S}_\kappa}{\partial \kappa} & \frac{\partial \mathcal{S}_\kappa}{\partial \epsilon} \\ \frac{\partial \mathcal{S}_\epsilon}{\partial \kappa} & \frac{\partial \mathcal{S}_\epsilon}{\partial \epsilon} \end{pmatrix} \\
\mathcal{D}_{\kappa,\epsilon} &= \begin{pmatrix} \frac{1}{\nu_{eff}/C_\kappa} & 0 \\ 0 & \frac{1}{\nu_{eff}/C_\epsilon} \end{pmatrix}
\end{aligned} \tag{28}
$$

$$
\begin{aligned}
J_{\kappa,\epsilon} &= \mathcal{D}^{-1}_{\kappa,\epsilon} C_{\kappa,\epsilon} \\
\mathit{\Lambda}_{\kappa,\epsilon} &= \begin{pmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{pmatrix} = \text{Eigenvalues}\Big\{ J_{\kappa,\epsilon} \Big\}
\end{aligned}
$$

If the eigenvalues are complex we take only the real part of them.

- Some comments about the change of the type of equations according to the type of eigenvalues of the reaction matrix jacobians. For example real or complex and if it is real, is possitive or negative. The relation between the type of eigenvalues and some upper bound in the turbulent length scale to be simulated.

3.3 Improving the convergence rate by a fully implicit treatment of the whole system

- cutoff function definition

The *Cutoff function* is useful to avoid the consequences of having negative values for κ and ϵ in the computation of the turbulent viscosity and the $\kappa - \epsilon$ production terms. It may be defined as

$$
\begin{aligned}
\mathit{\Xi}(x, \varepsilon) &= \begin{cases} \frac{1+1/2e^r}{1+\frac{1}{6}r^2}\varepsilon & \text{for } |r| < 10^{-7} \\ \frac{x-\varepsilon}{1-e^{-2r}} + \varepsilon & \text{for } r > 10^{-7} \\ \frac{(x-\varepsilon)e^{2r}}{e^{2r}-1} + \varepsilon & \text{for } r < -10^{-7} \end{cases} \\
r &= \frac{x}{\varepsilon} - 1
\end{aligned} \tag{29}
$$

Adding this cutoff function to the computation of the turbulent viscosity and the source terms they are rewritten as:

$$
\begin{aligned}
\nu_t &= C_\mu \frac{\mathit{\Xi}(\kappa, \varepsilon)^2}{\mathit{\Xi}(\epsilon, \varepsilon)} \\
P_\kappa &= 2\nu_t \boldsymbol{\epsilon}(\boldsymbol{u}) : \boldsymbol{\epsilon}(\boldsymbol{u}) \\
\mathcal{S}_\kappa &= P_\kappa - \mathit{\Xi}(\epsilon, \varepsilon) \\
\mathcal{S}_\epsilon &= \frac{\mathit{\Xi}(\epsilon, \varepsilon)}{\mathit{\Xi}(\kappa, \varepsilon)} (C_1 P_\kappa - C_2 \mathit{\Xi}(\epsilon, \varepsilon))
\end{aligned} \tag{30}
$$

with the derivatives of the cut-off function written as:

$$\frac{\mathrm{d}\mathit{\Xi}}{\mathrm{d}x} = \begin{cases} \frac{1/2e^r}{(1+\frac{1}{6}r^2)^2}(1+\frac{1}{6}r^2-\frac{1}{3}r) & \text{for } |r| < 10^{-7} \\ \frac{1}{1-e^{-2r}}(1-\frac{2re^{-2r}}{1-e^{-2r}}) & \text{for } r > 10^{-7} \\ \frac{e^{2r}}{e^{2r}-1}(1-\frac{2r}{e^{2r}-1}) & \text{for } r < -10^{-7} \end{cases} \tag{31}$$

According to 28 the eigenvalues problem should be updated taking into account the introduction of the cutoff function. Defining $G = 2C_\mu \boldsymbol{\epsilon}(\boldsymbol{u}) : \boldsymbol{\epsilon}(\boldsymbol{u})$ to simplify the notation the $C_{\kappa,\epsilon}$ matrix is written as:

$$\begin{aligned} C_{\kappa,\epsilon} &= \begin{pmatrix} \frac{\partial \mathcal{S}_\kappa}{\partial \kappa} & \frac{\partial \mathcal{S}_\kappa}{\partial \epsilon} \\ \frac{\partial \mathcal{S}_\epsilon}{\partial \kappa} & \frac{\partial \mathcal{S}_\epsilon}{\partial \epsilon} \end{pmatrix} \\ &= \begin{pmatrix} 2G\frac{\mathit{\Xi}(\kappa,\varepsilon)}{\mathit{\Xi}(\epsilon,\varepsilon)}\frac{\partial \mathit{\Xi}(\kappa,\varepsilon)}{\partial \kappa} & (-G(\frac{\mathit{\Xi}(\kappa,\varepsilon)}{\mathit{\Xi}(\epsilon,\varepsilon)})^2-1)\frac{\partial \mathit{\Xi}(\epsilon,\varepsilon)}{\partial \epsilon} \\ (C_1 G + C_2(\frac{\mathit{\Xi}(\epsilon,\varepsilon)}{\mathit{\Xi}(\kappa,\varepsilon)})^2)\frac{\partial \mathit{\Xi}(\kappa,\varepsilon)}{\partial \kappa} & (-2C_2\frac{\mathit{\Xi}(\epsilon,\varepsilon)}{\mathit{\Xi}(\kappa,\varepsilon)}\frac{\partial \mathit{\Xi}(\epsilon,\varepsilon)}{\partial \epsilon}) \end{pmatrix} \end{aligned} \tag{32}$$

- implicit treatment of wall law function in the boundary conditions via Lagrange multipliers.

The numerical treatment of the wall-law boundary conditions 9 and 11 is a special topic that deserves some attention. The main reason is the great influence that this boundary condition exerts over the global convergence of the algorithm. In 14 the Neumann boundary conditions are introduced in the $\int_{\mathit{\Gamma}_h} \boldsymbol{w}^h \cdot \boldsymbol{h}^h d\mathit{\Gamma}$ term in a natural way. According to the definition of the boundary conditions for the incompressible Navier-Stokes equations given by 7 and having in mind that the wall law function is applied to the momentum equations as a Neumann boundary condition (10), then the residual contribution coming from this term looks like:

$$\begin{aligned} \int_{\mathit{\Gamma}_h} \boldsymbol{w}^h \cdot \boldsymbol{h}^h d\mathit{\Gamma} &= \int_{\mathit{\Gamma}_h - \mathit{\Gamma}_{\text{wall}}} \boldsymbol{w}^h \cdot \boldsymbol{h}^h d\mathit{\Gamma}_h + \int_{\mathit{\Gamma}_{\text{wall}}} \boldsymbol{w}^h \cdot \boldsymbol{h}^h d\mathit{\Gamma} wall = \\ &= \int_{\mathit{\Gamma}_h - \mathit{\Gamma}_{\text{wall}}} \boldsymbol{w}^h \cdot \boldsymbol{n} \cdot \boldsymbol{\sigma} d\mathit{\Gamma}_h + \int_{\mathit{\Gamma}_{\text{wall}}} \boldsymbol{w}^h \cdot g(\boldsymbol{u} - \boldsymbol{u}_{\text{wall}}) d\mathit{\Gamma} \end{aligned} \tag{33}$$

The fully implicit treatment of the whole system requires the derivation of the $\int_{\mathit{\Gamma}_{\text{wall}}}$ integral term in 33 respect to the state variables, specifically $\boldsymbol{u}^h$.

$$
\begin{aligned}
\frac{\partial \boldsymbol{R}_i}{\partial \boldsymbol{u}_j} &= \dots - \int_{\Gamma_{\mathrm{wall}}} \boldsymbol{w}_i^h \cdot (g + g' \underbrace{\frac{\partial |(\boldsymbol{u} - \boldsymbol{u}_{\mathrm{wall}})|}{\partial \boldsymbol{u}}}_{\frac{(\boldsymbol{u}-\boldsymbol{u}_{\mathit{wall}})}{|(\boldsymbol{u}-\boldsymbol{u}_{\mathit{wall}})|}} (\boldsymbol{u} - \boldsymbol{u}_{\mathrm{wall}}))\, \underbrace{\frac{\partial \boldsymbol{u}}{\partial \boldsymbol{u}_j}}_{\boldsymbol{w}_j^h} d\Gamma = \\
&= \int_{\Gamma_{\mathrm{wall}}} \boldsymbol{w}_i^h \cdot (g + g' |\boldsymbol{u} - \boldsymbol{u}_{\mathrm{wall}}|) \boldsymbol{w}_j^h d\Gamma \\
g' &= \frac{\partial g}{\partial |(\boldsymbol{u} - \boldsymbol{u}_{\mathrm{wall}})|} = (2 f(y^+) \frac{\partial u_*}{\partial |(\boldsymbol{u} - \boldsymbol{u}_{\mathit{wall}})|} - 1) \frac{g}{|(\boldsymbol{u} - \boldsymbol{u}_{\mathit{wall}})|}
\end{aligned}
\tag{34}
$$

In 34 $\frac{\partial u_*}{\partial |(\boldsymbol{u}-\boldsymbol{u}_{\mathit{wall}})|}$ may be computed from 9 straightforwardly.

The main difficulty in the fully implicit treatment of wall-law boundary conditions arises with the Dirichlet fixation of κ and ϵ from 11 because this fixation depends on the unknown velocity vector at this boundary. In general this kind the boundary conditions are numerically solved via an explicit formulation where the value of the current κ and ϵ at the wall are fixed to the corresponding state variable in the last time step. Even though this strategy is very simple to adapt to an existing code its main disadvantage is related to the poor convergence rate reached with it. In this work we propose a strategy based on an implicit treatment of this boundary condition. To do this we use Lagrange multipliers in order to add the extra equations needed to impose 11. For each node where we impose the wall law function we add a *ficticious node* where we solve for the fluxes needed to balance the unbalanced fluxes in the original nodes after imposing over them the satisfaction of the 11.

The concept of elemset (see Petscfem documentation in http://minerva.arcride.edu.ar/petscfem/petscfem) allows to add the contribution of this nonlinear restriction in an addittive way as:

$$
\begin{aligned}
R_a &= R_a + \Lambda U_\alpha LSF \\
R_\alpha &= R_{\mathrm{wall}}^{\kappa-\epsilon} - LDF\ \ LRF\ \ U_\alpha \\
K_{a,\alpha}(:, 1:2) &= -LSF\ \ \Lambda \\
K_{\alpha,a}(1:2, :) &= -K_{\mathrm{wall}}^{\kappa-\epsilon} \\
K_{\alpha,\alpha}(1:2, 1:2) &= LDF \\
\Lambda &= \begin{pmatrix} 0 & 0 \\ \vdots & \vdots \\ 1 & 0 \\ 0 & 1 \end{pmatrix} \quad (ndof \times 2)
\end{aligned}
\tag{35}
$$

where a is the original wall node and α is the corresponding ficticious node. Each ficticious node adds two more equations and two more degree of freedom, i.e:

$$U_\alpha = \begin{pmatrix} \lambda_\kappa \\ \lambda_\epsilon \end{pmatrix} \tag{36}$$

the two Lagrange multipliers containing the fluxes neded to balance the κ and ϵ transport equations having imposed the wall-law values for the turbulent kinetic energy and the dissipation rate. R_a and R_α are the residual vectors of these two nodes and $R^{\kappa-\epsilon}_{wall}$ is the redidual of the added wall law equations defined as:

$$R^{\kappa-\epsilon}_{wall} = \begin{pmatrix} \kappa - \kappa_{\mathrm{wall}} \\ \epsilon - \epsilon_{\mathrm{wall}} \end{pmatrix}$$

$$K^{\kappa-\epsilon}_{wall} = \begin{pmatrix} -\frac{1}{|u|}\frac{\partial \kappa_{\mathrm{wall}}}{\partial |u|}(\boldsymbol{u} - \boldsymbol{u}_{\mathrm{wall}}) & 0 & 1 & 0 \\ -\frac{1}{|u|}\frac{\partial \epsilon_{\mathrm{wall}}}{\partial |u|}(\boldsymbol{u} - \boldsymbol{u}_{\mathrm{wall}}) & 0 & 0 & 1 \end{pmatrix} \tag{37}$$

LRF is named *Lagrange Residual Factor*. Using Lagrange multipliers leads to diagonal null terms, which can cause zero pivots when using direct methods. With this option a small term is added to the diagonal in order to fix this. The term is added only in the Jacobian or also in the residual (which results would be non-consistent).

LDF is named *Lagrange diagonal factor*. A diagonal term proportional may be also entered in the residual. If this is so using *LRF* $= 1$ then the method is “non-consistent”, i.e. the restriction is not exactly satisfied by the non-linear scheme is exactly Newton-Raphson. If not *LRF* $= 0$ then the restriction is consistently satisfied but with a non exact Newton-Raphson.

LSF is named *lagrange scale factor*. Using Lagrange multipliers can lead to bad conditioning, which causes poor convergence with iterative methods or amplification of rounding errors. This factor scales the columns in the matrix that correspond to the lagrange multipliers and can help in better conditioning the system.

4 SOLVER

PETSc-FEM is a general purpose, parallel, multi-physics FEM program for CFD applications based on PETSc, therefore all the solvers and preconditioners included in PETSc distribution are available. However, PETSc-FEM includes the possibility of using another method based on domain decomposition that is briefly described in the next section. For more details visit the Web site http://minerva.arcride.edu.ar/petscfem/petscfem .

4.1 IISD - An algebraic solver based on domain decomposition

IISD (for *"Interface Iterative – Sub-domain Direct"*) is an hybrid solver scheme using a direct solver inside each subdomain and an iterative solver only on the interface among the subdomains. The subdomain partition is not only done at processor level, also it is possible to split the processor domain itself in several subdomains, making this strategy extensible to scalar computation too. Chossing the number of subdomains it is possible to cover from pure direct method in one side to full iterative method on the other side. This strategy enhances the convergence rate and it can be seen as a parallel direct method.

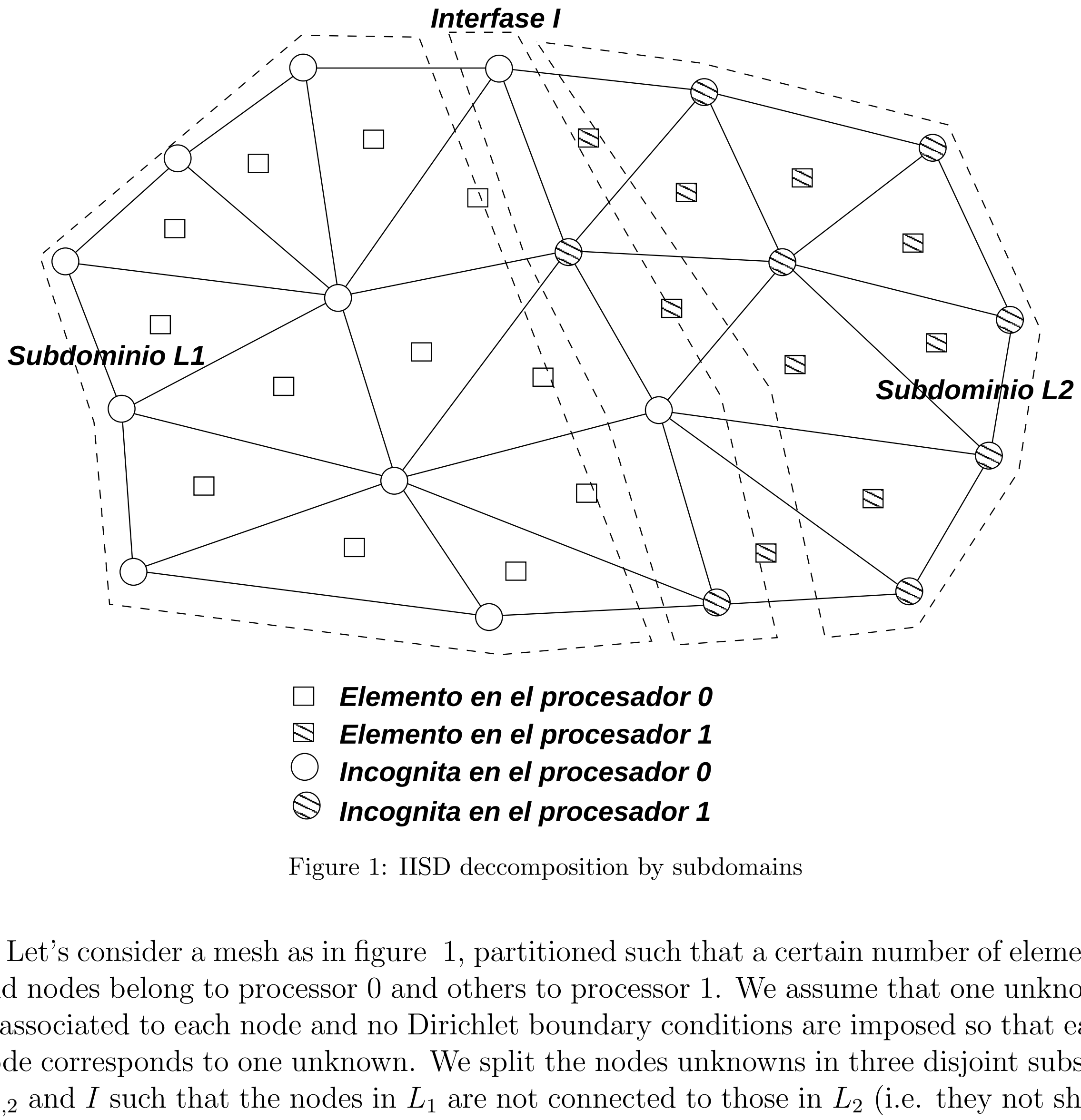
Interfase I
Subdominio L1
Subdominio L2
□ Elemento en el procesador 0
▨ Elemento en el procesador 1
○ Incognita en el procesador 0
◍ Incognita en el procesador 1

Figure 1: IISD deccomposition by subdomains

Let's consider a mesh as in figure 1, partitioned such that a certain number of elements and nodes belong to processor 0 and others to processor 1. We assume that one unknown is associated to each node and no Dirichlet boundary conditions are imposed so that each node corresponds to one unknown. We split the nodes unknowns in three disjoint subsets $L_{1,2}$ and I such that the nodes in L_1 are not connected to those in L_2 (i.e. they not share an element, and then, the FEM matrix elements $A_{i,j}$ and $A_{j,i}$ with $i \in L_1$ and $j \in L_2$ are null. The matrix is split in blocks as follows

$$
\begin{aligned}
\boldsymbol{A} &= \left[\begin{array}{cc} \boldsymbol{A}_{LL} & \boldsymbol{A}_{LI} \\ \boldsymbol{A}_{IL} & \boldsymbol{A}_{II} \end{array}\right] \\
\boldsymbol{A}_{LL} &= \left[\begin{array}{cc} \boldsymbol{A}_{11} & 0 \\ 0 & \boldsymbol{A}_{11} \end{array}\right] \\
\boldsymbol{A}_{LI} &= \left[\begin{array}{cc} \boldsymbol{A}_{1I} & \boldsymbol{A}_{2I} \end{array}\right] \\
\boldsymbol{A}_{IL} &= \left[\begin{array}{c} \boldsymbol{A}_{I1} \\ \boldsymbol{A}_{I2} \end{array}\right]
\end{aligned}
\tag{38}
$$

Now consider the system of equations

$$\boldsymbol{A}\boldsymbol{x} = \boldsymbol{b} \tag{39}$$

which is split as

$$
\begin{aligned}
\boldsymbol{A}_{LL}\,\boldsymbol{x}_L + \boldsymbol{A}_{LI}\,\boldsymbol{x}_I &= \boldsymbol{b}_L \\
\boldsymbol{A}_{IL}\,\boldsymbol{x}_L + \boldsymbol{A}_{II}\,\boldsymbol{x}_I &= \boldsymbol{b}_I
\end{aligned}
\tag{40}
$$

Now consider eliminating $\boldsymbol{x}_L$ from the first equation and replacing in the second so that, we have an equation for $\boldsymbol{x}_I$

$$
\begin{aligned}
(\boldsymbol{A}_{II} - \boldsymbol{A}_{IL}\,\boldsymbol{A}_{LL}^{-1}\,\boldsymbol{A}_{LI})\,\boldsymbol{x}_I &= (\boldsymbol{b}_I - \boldsymbol{A}_{IL}\,\boldsymbol{A}_{LL}^{-1}\,\boldsymbol{b}_L) \\
\tilde{\boldsymbol{A}}\,\boldsymbol{x}_I &= \tilde{\boldsymbol{b}}_I
\end{aligned}
\tag{41}
$$

We consider solving this system of equations by an iterative method such as GMRES, for instance. For such an iterative method, we have only to specify how to compute the modified right hand side $\tilde{\boldsymbol{b}}$ and also how to compute the matrix-vector product $\boldsymbol{y} = \tilde{\boldsymbol{A}}\,\boldsymbol{x}$. Computing the matrix product involves the following steps

1. Compute $\boldsymbol{y} = \boldsymbol{A}_{II}\,\boldsymbol{x}$
2. Compute $\boldsymbol{w} = \boldsymbol{A}_{LI}\,\boldsymbol{x}$
3. Solve $\boldsymbol{A}_{LL}\,\boldsymbol{z} = \boldsymbol{w}$ for $\boldsymbol{z}$
4. Compute $\boldsymbol{v} = \boldsymbol{A}_{IL}\,\boldsymbol{z}$
5. Add $\boldsymbol{y} \leftarrow \boldsymbol{y} - \boldsymbol{v}$

involving three matrix products with matrices $\boldsymbol{A}_{II}$, $\boldsymbol{A}_{IL}$ and $\boldsymbol{A}_{LI}$ and to solve the system with $\boldsymbol{A}_{LL}$. As the matrix $\boldsymbol{A}_{LL}$ has no elements connecting unknowns in different processors the solution system may be computed very efficiently in parallel.

5 NUMERICAL EXAMPLE

We have chosen the square cylinder benchmark in order to validate our computational implementation. This physical problem is defined by the flow past through a cylinder of square cross-section, placed centrally in a channel.The wake of the square cylinder is unsteady and driven by the shed vortices. This type of problem is of great importance in engineering applications and present numerical difficulties in the solution.

5.1 Computational domain and boundary condition

The computational domain is sketched in the figure 3, when the dimensions of the channel are the following: H=8,B=1 and L=32.The cylinder is placed at a distance of Lr=8.0 unit from the inlet.The mesh employed in the computation is unstructured, with 12000 triangular elements. A global view and a close up of the mesh is shown in figures 3 4 where we adopt a similar partition to that published in[5] in order to do some preliminar validation. In the close up figure is viewed that 10 elements by each side of the square cylinder are placed and we think that this coarse local mesh is not adequate to capture the main fluid patterns in the vecinity of this bluff body. So, future computations should be done over a finer mesh in the cylinder region.

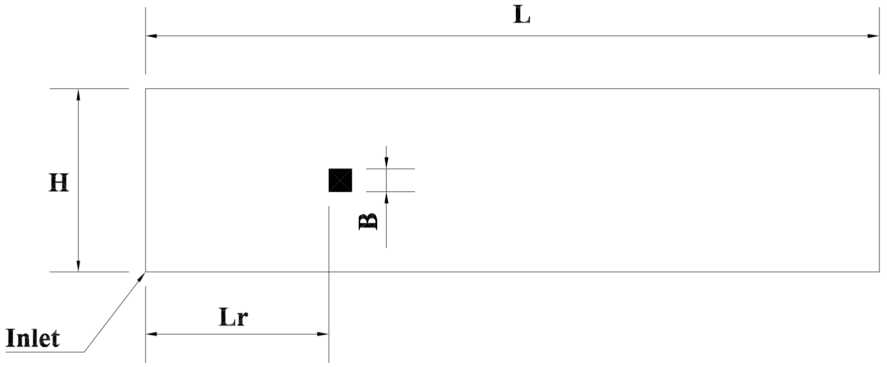

Figure 2: Geometry

At the inlet the flow enters with uniform velocity u_{av} and the prescribed turbulence intensity defined as $I = \sqrt{u_i'^2/2}/u_{\mathrm{av}}$ is set to 10 percent at this plane. The eddy viscosity at the inflow plane is specified as $\nu_t/\nu_t = 10$, so the value of ϵ is computed according to the equation 4.

The free-slip boundary conditions have been applied at the top and bottom surface. Wall function treatment has been used for all the solid boundaries. At the outlet plane

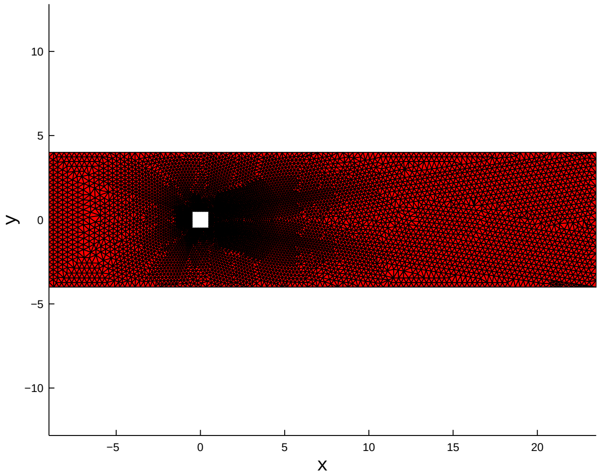

Figure 3: Global mesh

the pressure and the y component of the velocity is fixed to 0 at the outflow plane.

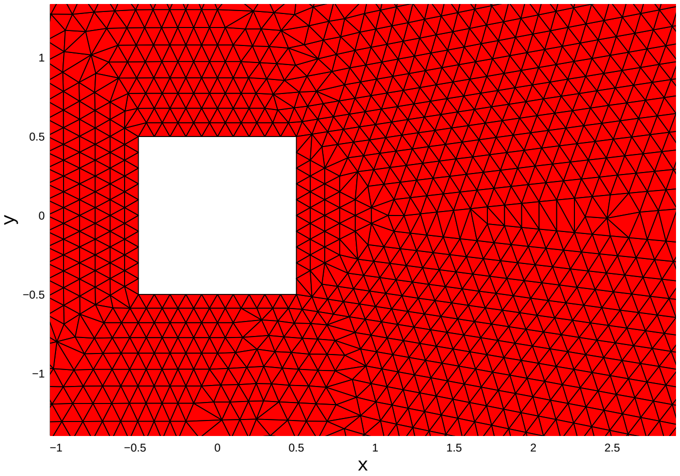

Figure 4: Close up of the mesh

5.2 Comparison with experiment

In the figure 5 and 6 we showed a comparison of the time-averaged streamwise velocity profile at the different downstream locations , namely (x=1 and 5) for our $\kappa-\epsilon$ turbulence model implementation relative to some experimental measurements. The experimental data of Lyn et.al[6] has also been plotted in this figure. The time-averaged component of the velocities have been obtained by integrating the instantaneous fields over a long duration, with 20 shedding cycles.

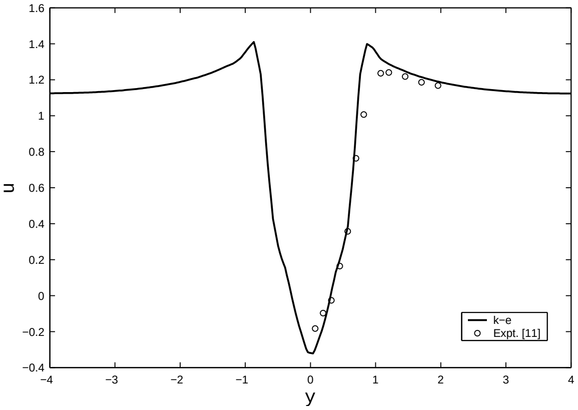

Figure 5: Time-averaged streamwise velocity profile at x=1

At the location x=1 we may note that the agreement is quite good. The main difference is presented at the center of the channel, where the shape of the velocities profile is the same, but exist some offset of the values. We suspect that the main discrepancy could be due to a high blockage due to a small width between top and bottom channel walls. Far downstream the comparison reveals more discrepancy, possibly caused by the same reason.

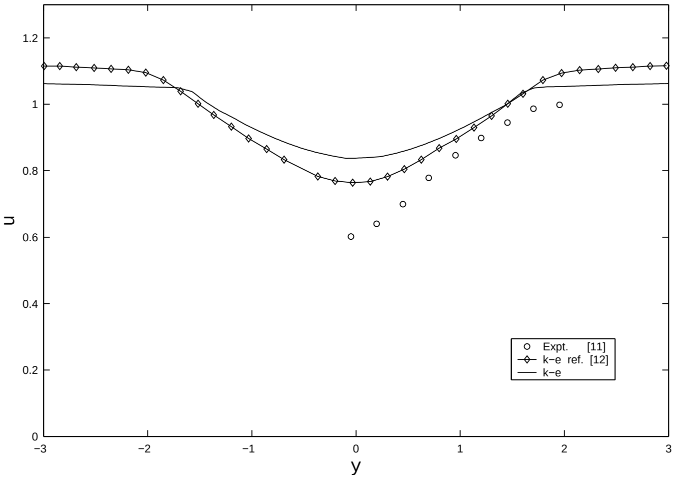

Figure 6: Time-averaged streamwise velocity profile at x=5

Fig 7 shows the time-average y-component of the velocity compared with the experiments of Lyn et .al^6 at the plane x=5.

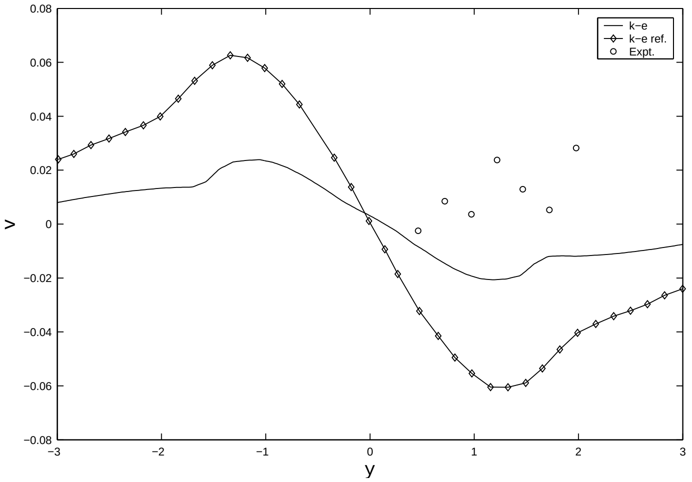

Figure 7: Time-averaged traverse velocity profile at x=5

6 CONCLUSIONS

This work focus on the main numerical drawbacks found in the simulation of turbulent flows. Some first results reveals that some addtional stabilization is necessary for $\kappa - \epsilon$ production terms and some control strategy for the wall law boundary condition is crucial. In this preliminar version of our implementation we have adopted for the former item an extra numerical diffusion term based on some earlier background on solving advective-reactive-diffusive systems of equations. For the later item we have done some numerical continuation in the y_{wall} value in order to get good convergence rate in the Newton scheme. Much more work on these items should be done in order to get more robustness in the computational code.

7 ACKNOWLEDGMENT

This work has received financial support from *Consejo Nacional de Investigaciones Científicas y Técnicas* (CONICET, Argentina), *Banco Interamericano de Desarrollo* (BID) and *Universidad Nacional del Litoral* through grants CONICET PIP 198/98, ANPCyT PICT51 and CAI+D UNL 94/95. We made extensive use of freely distributed software as *Linux* OS, MPI, PETSc, Newmat, Visual3, GID and many others.

REFERENCES

[1] MPI Forum, http://www.mpi-forum.org/docs/docs.html

[2] The Beowulf Project, http://www.beowulf.org

[3] PETSc Scientific Library, http://www-unix.mcs.anl.gov/petsc/

[4] T. Tezduyar, S. Mittal, S. Ray, R. Shih, “Incompressible flow computations with stabilized bilinear and linear equal order interpolation velocity-pressure elements”, *Comp. Meth. Applied Mechanics and Engineering*, 95, 1992.

[5] A. K. Saha, G. Biswas and K. Muralidhar,"Two-Dimensional Study of the Turbulent Wake Behing a Square Cylinder Subject to Uniform Shear",J.Fluids Eng.,123,pp.,595-603,2000.

[6] D. A. Lyn, S. Einav, W. Rodi, and J. H.Park,"A Laser-Doppler Velocimetry Study of Emsemble-Averaged Characteristic of Turbulent Near Wake of Square Cylinder,"Phys.Fluids,7,pp.1841-1867,1995.

[7] S. Idelsohn, M. Storti, N. Nigro and G. Buscaglia, "A Petrov-Galerkin formulation for advection-reaction-diffusion problems", Computer Methods in Applied Mechanics and Engineering, 136, pp. 27-46, 1996.

[8] J.O. Hinze,"Turbulence",McGraw-Hill Book Company ,Inc.,1959.

[9] H. Tennekes, J. Lumley,"A firts course in Turbulence", The MIT Press,1972.

[10] D.C. Wilcox,"Turbulence Modeling for CDF ",DCW Industries,Inc.,2000.